AppleScript Help

Basisonderwerpen

Informatie over AppleScript

Bij het werken met de Mac zult u merken dat u bepaalde taken steeds opnieuw uitvoert. Met behulp van AppleScript kunt u scripts en programma's schrijven waarmee u deze taken kunt automatiseren.

AppleScript is een krachtige en veelzijdige scripttaal die is ingebouwd in Mac OS X. U kunt AppleScript gebruiken om verkorte procedures aan te maken, terugkerende taken te automatiseren of zelfs maatwerkprogramma's te schrijven die u veel tijd kunnen besparen. Met AppleScript kunt u ook scriptgeschikte programma's besturen.

Een AppleScript-script bestaat uit commando's die op Engels lijken waarmee handelingen kunnen worden uitgevoerd op objecten. Enkele voorbeelden van handelingen zijn openen, sluiten of de naam opvragen. Enkele voorbeelden van een object zijn een programma, een venster en een voorkeursinstelling.

Met een script kunnen een of meer programma's worden aangestuurd. Een programma is scriptgeschikt als voor het programma de termen zijn gedefinieerd die in de scripts kunnen worden gebruikt. OS X wordt geleverd met verschillende scriptgeschikte programma's.

Met scripts kunt u ook programma's via een netwerk aansturen, zodat u taken op meerdere computers kunt automatiseren.

Het is zelfs zo dat u AppleScript hebt gebruikt zonder dat u daar iets van gemerkt hebt. Veel functies in het systeem en in programma's zijn eigenlijk AppleScript-scripts die worden aangeroepen vanuit een programma. Als u bijvoorbeeld in dit Help-systeem klikt op een koppeling waarmee een programma of bestand wordt geopend, wordt die handeling uitgevoerd door een AppleScript-script.

VERWANTE ONDERWERPEN

Informatie over de manier waarop AppleScript werkt
Informatie over AppleScript-editor
Informatie over de AppleScript-taal
Informatie over AppleScript op het internet
AppleScript Studio-ontwikkelomgeving

Basisonderwerpen

Informatie over AppleScript-editor

AppleScript-editor is een gebruiksvriendelijk scriptprogramma waarmee u AppleScript-scripts kunt lezen, schrijven, opnemen en bewaren.

Enkele kenmerken van AppleScript-editor zijn:

- Een gecombineerde, gebruiksvriendelijke interface
- De mogelijkheid om tekst in scripts te zoeken en te vervangen
- Automatische regelterugloop bij lange instructies

- De mogelijkheid om meerdere malen 'Herstel' te kiezen
- Scripts in een contextueel menu, zodat u sneller scripts kunt schrijven
- Ondersteuning voor programmavoorzieningen
- Een bibliotheekpalet dat toegang biedt tot programmascripts
- Een scriptassistent die u helpt instructies te voltooien
- Vensters 'Resultaatgeschiedenis' en 'Eventgeschiedenis' waarin u kunt nagaan wat er precies gebeurt in een script

VERWANTE ONDERWERPEN

Informatie over de AppleScript-taal
AppleScript Studio-ontwikkelomgeving

Basisonderwerpen

AppleScript-scripts aanmaken

U hoeft helemaal geen programmeerkennis te hebben om te kunnen werken met AppleScript. Het is zelfs zo dat u zelf geen letter script hoeft te schrijven. OS X wordt geleverd met honderden handige AppleScript-scripts die u direct kunt gebruiken. Op het internet kunt u bovendien nog talloze andere scripts vinden.

Als u iets beter wilt leren werken met AppleScript, kunt u een bestaand script nemen en dit met minimale wijzigingen aanpassen aan uw behoeften.

U kunt natuurlijk het meeste uit AppleScript halen als u zelf scripts gaat schrijven. Het voordeel hiervan is dat u handige en tijdbesparende programma's kunt ontwikkelen waarmee u nog efficiënter kunt werken.

Met AppleScript-editor kunt u bestaande scripts aanpassen of zelf scripts schrijven. AppleScript-editor heeft een gebruiksvriendelijke interface voor het schrijven en bewerken van scripts. Alle voorbeelden in de AppleScript Help zijn gemaakt met AppleScript-editor.

- Open AppleScript-editor.

  Typ de volgende tekst in het venster van AppleScript-editor.

  ```
  say "Welkom bij AppleScript!"
  ```

  Klik op 'Voer uit'.

VERWANTE ONDERWERPEN

Informatie over AppleScript-editor
AppleScript Studio-ontwikkelomgeving

Basisonderwerpen

Informatie over de manier waarop AppleScript werkt

De essentie van AppleScript is het verzenden van speciale berichten die “Apple events” worden genoemd. Wanneer u een script uitvoert, verstuurt AppleScript Apple events naar het systeem of programma's die de commando's in het script

vervolgens uitvoeren.

In veel gevallen sturen de programma's of het systeem een resultaat terug naar AppleScript. Als in een script bijvoorbeeld de naam van een object wordt opgevraagd bij een programma, stuurt het programma de objectnaam terug. Als het commando niet kan worden uitgevoerd, retourneert het programma of AppleScript mogelijk een foutmelding. De resultaten van commando's kunt u bekijken in AppleScript-editor.

Over het algemeen hoeft u niets te weten over de precieze werking van Apple events. Het afhandelen van events wordt onzichtbaar op de achtergrond uitgevoerd door AppleScript. U hoeft alleen maar scripts te schrijven en uit te voeren.

VERWANTE ONDERWERPEN

Informatie over AppleScript
Informatie over de AppleScript-taal

Basisonderwerpen

Informatie over applets

Een script dat wordt bewaard als een programma wordt een AppleScript-programma of applet genoemd. Applets worden automatisch gestart als ze worden geopend, waarbij de commando's in het script worden uitgevoerd. Applets kunnen kleine functies zijn met slechts een paar commando's, maar ook complexe programma's met duizenden regels code.

Applets kunnen worden bewaard als bewerkbare programma's of als programma's die uitsluitend kunnen worden uitgevoerd. Een bewerkbare applet bevat ook de tekst van het script en kan daardoor steeds opnieuw worden bewerkt en gecompileerd. Als u een applet wilt bewerken, sleept u de applet naar het symbool van AppleScript-editor. Wanneer u een script bewaart dat uitsluitend kan worden uitgevoerd, wordt de bewerkbare tekst verwijderd en kan het bestand niet meer worden bewerkt of gecompileerd.

Als u een applet wilt wijzigen die is opgeslagen als een bewerkbaar programma, opent u het programma in AppleScript-editor of een ander programma voor het bewerken van AppleScripts. Applets vereisen een "on run"-handler, die het script gaat uitvoeren zodra de applet wordt geopend.

- In dit voorbeeld ziet u een 'on run'-handler.

```
on run
        --eerste commando
        --tweede commando
end run
```

VERWANTE ONDERWERPEN

Informatie over droplets
AppleScript-bestandstypen

Basisonderwerpen

Informatie over gecompileerde scripts

Een gecompileerd script is een AppleScript-bestand in een bewerkbare vorm dat u kunt lezen en wijzigen in AppleScript-editor of in een ander programma voor het bewerken van AppleScripts.

Wanneer u een gecompileerd script opent, wordt het script weergegeven in AppleScript-editor maar worden de commando's niet automatisch uitgevoerd. Op deze manier kunt u het script wijzigen zonder dat het daadwerkelijk wordt uitgevoerd. Vervolgens kunt u het gewijzigde script compileren om alle tekst in het bestand in te lezen en te converteren naar AppleScript-code. Het gecompileerde script kan worden uitgevoerd. Een gecompileerd script heeft de bestandsnaamextensie .scpt.

Wanneer u het bestand bewaart als een gecompileerd script, wordt de originele tekst in het bestand samen met de code opgeslagen. Het gecompileerde script kan daardoor steeds opnieuw worden bewerkt.

VERWANTE ONDERWERPEN

Informatie over applets
Informatie over droplets
AppleScript-bestandstypen

Basisonderwerpen

Informatie over droplets

Een droplet is een AppleScript-script dat wordt uitgevoerd wanneer u bestanden of mappen naar het symbool van het script sleept. De handelingen in het script worden vervolgens uitgevoerd op deze bestanden of mappen.

Net als een applet kan een droplet worden bewaard als een bewerkbaar programma of als een programma dat uitsluitend kan worden uitgevoerd. Een bewerkbare droplet bevat ook de tekst van het script en kan daardoor steeds opnieuw worden bewerkt en gecompileerd. Als u een droplet wilt bewerken, sleept u de droplet naar het Scripteditor-symbool. Wanneer u een script bewaart dat uitsluitend kan worden uitgevoerd, wordt de bewerkbare tekst verwijderd en kan het bestand niet meer worden bewerkt of gecompileerd.

Droplets vereisen een "on open"-handler, die het script gaat uitvoeren zodra de droplet wordt geopend. De "on open"-handler bevat een parameter met een lijst van de onderdelen die naar de droplet kunnen worden gesleept. Deze lijst wordt gecontroleerd met behulp van bepaalde commando's in de droplet, waarna andere commando's handelingen uitvoeren op de onderdelen. Tenzij de droplet slechts op één onderdeel tegelijk kan worden uitgevoerd, bevat de droplet een lus die de handelingen in het script uitvoert op het eerste onderdeel in de lijst en ze vervolgens herhaalt voor alle overige onderdelen in de lijst.

- Hier ziet u een voorbeeld van een droplet-handler met 'on open' en de parameterlijst.

```
on open (de_gesleepte_onderdelen)
        --eerste commando
        --tweede commando
end open
```

VERWANTE ONDERWERPEN

Informatie over applets
Informatie over gecompileerde scripts
AppleScript-bestandstypen

Basisonderwerpen

AppleScript-woordenboeken gebruiken

Als u wilt weten welke commando’s en objecten u kunt gebruiken om een programma aan te sturen, raadpleegt u het woordenboek van het programma. Een woordenboek bevat een reeks termen die door een programma kunnen worden verwerkt.

U kunt het woordenboek van een programma in AppleScript-editor bekijken door het programma in het bibliotheekvenster te selecteren of door het programmasymbool naar het symbool van AppleScript-editor te slepen.

De termen in een woordenboek zijn ingedeeld in groepen met verwante termen. Deze groepen worden "suites" genoemd. Commando's en objecten worden vet weergegeven.

---

1. Kies 'Venster' > 'Bibliotheek' in het programma AppleScript-editor.
2. Klik dubbel op de naam van het programma in het bibliotheekvenster.
3. Selecteer een suite.

   De termen in de suite en de bijbehorende beschrijvingen worden weergegeven.

De AppleScript-taal bestaat uit verschillende categorieën:

- Suites: Groepen verwante termen. In het woordenboek van AppleScript-editor worden suites in de eerste kolom weergegeven en aangeduid met een oranje letter 'S'.
- Commando's: Hiermee wordt verzocht om een handeling of resultaat. Commando's, die ook wel "events" worden genoemd, verschijnen in de tweede kolom en worden aangeduid met een blauwe letter 'C'.
- Klassen: Scriptgeschikte objecten. De beschrijving van een klasse bestaat uit de kenmerken en elementen die tot de klasse behoren. Klassen verschijnen in de tweede kolom en worden aangeduid met een lavendelkleurige letter 'C'.
- Kenmerken: Objecten die waarden bevatten. Kenmerken verschijnen in de derde kolom en worden aangeduid met een lavendelkleurige letter 'P'.
- Elementen: Objecten die deel uitmaken van andere objecten. Een alinea kan bijvoorbeeld vele woorden bevatten. Elementen kunnen in de derde kolom worden weergegeven en worden aangeduid met een oranje letter 'E'.

VERWANTE ONDERWERPEN

Klassendefinities
Commandodefinities
AppleScript gebruiken om opdrachten te geven aan programma's

OPEN HET VOLGENDE

AppleScript Language Guide: Commands Reference
AppleScript Language Guide: Class Reference

Basisonderwerpen

Problemen oplossen

Als u problemen ondervindt bij het gebruik van AppleScript, kan de informatie in de volgende onderwerpen mogelijk uitkomst bieden.

Foutmelding bij klikken op 'Annuleer' in venster dat is geopend met 'choose from list'

Een mapactie wordt onverwacht geactiveerd

Een schijf wordt niet door een script herkend

Klassendefinities

Een klassendefinitie beschrijft de kenmerken (“property's”) en elementen van een scriptgeschikt object. U kunt een klasse vergelijken met een zelfstandig naamwoord in een zin en de bijbehorende klassendefinitie met de definitie van het zelfstandige naamwoord in een woordenboek.

Een klassendefinitie in een AppleScript-woordenboek begint altijd met de term waar het om gaat (deze wordt vet weergegeven), gevolgd door een korte beschrijving van deze term (op dezelfde regel).

Hierna volgt een lijst met alle elementen of objecten die het scriptgeschikte object vormen of ertoe behoren. Niet elke klassendefinitie bevat een lijst met elementen. Sommige scriptgeschikte objecten bevatten andere scriptgeschikte objecten, maar dat is niet altijd het geval.

De klassendefinitie eindigt met een lijst met kenmerken van het scriptgeschikte object en de bijbehorende waarden.

De informatie over een kenmerk begint altijd met de naam van het kenmerk, gevolgd door:

- Een beschrijving van de gegevensstructuur waarin de waarde van het kenmerk wordt bewaard, bijvoorbeeld "integer" (geheel getal), "point" (een lijst van twee cijfers), "string" (tekstgegevens) en "Boolean" (waar of niet waar).
- Een optionele typeaanduiding, [r/o], die aangeeft dat de waarde van het kenmerk alleen kan worden gelezen ("read-only", het kenmerk kan niet worden gewijzigd).
- Een korte beschrijving van het kenmerk.

Als een object een meervoudsvorm heeft, kunt u in een script naar meerdere objecten van hetzelfde type verwijzen.

Een object kan een lijst met elementen bevatten. Dit zijn andere objecten die zich in het object bevinden. De naam van een element kan in het script worden gebruikt en wordt vet weergegeven. De normale tekst die op de naam van het element volgt, beschrijft hoe in het script naar het element moet worden verwezen.

Objecten kunnen kenmerken ("property's") hebben. De naam van een kenmerk wordt vet weergegeven. De normale tekst die op de naam van het kenmerk volgt, beschrijft het type waarde dat het kenmerk bevat, bijvoorbeeld tekst (een "string"). Kenmerken kunnen een constante waarde of een reeks constante waarden hebben. Een script kan de waarde van een kenmerk opvragen en instellen. Instellen is niet mogelijk als de syntaxis van het kenmerk wordt gevolgd door '[r/o]' ('read-only' ofwel alleen lezen).

VERWANTE ONDERWERPEN

Commandodefinities
Objectdefinities
AppleScript-commando's
AppleScript-woordenboeken gebruiken

OPEN HET VOLGENDE

AppleScript Language Guide: Class Reference

Commandodefinities

Commando's zijn termen in AppleScript waarmee handelingen worden uitgevoerd. Commando's in een AppleScript-woordenboek staan in de middelste kolom en worden aangeduid met een blauwe letter 'C'. De definitie van een commando bestaat uit een korte beschrijving van wat het commando doet en de syntaxis. De syntaxis is de vorm waarin u het commando in een script moet gebruiken. De syntaxis kan uit verschillende delen bestaan.

Woorden die vet worden weergegeven, zijn de termen die u in een instructie gebruikt. U moet deze termen precies zo schrijven als ze in het woordenboek staan.

Woorden die met een blauwe kleur worden weergegeven, zijn gegevens die u opgeeft. Dit kan bijvoorbeeld een object zijn of een variabele die u hebt gedefinieerd.

Gedeelten tussen haakjes zijn optioneel. Deze hoeven niet te worden opgenomen in een instructie.

VERWANTE ONDERWERPEN

Klassendefinities
AppleScript-woordenboeken gebruiken
Eventdefinities
Objectdefinities

OPEN HET VOLGENDE

AppleScript Language Guide: Commands Reference

Eventdefinities

Een eventdefinitie beschrijft een werkwoord of commando waarmee een handeling op een scriptgeschikt object wordt gericht. Events zijn de handelingen of werkwoorden die objecten kunnen uitvoeren of waaraan ze kunnen bijdragen.

Een eventdefinitie begint altijd met de term waar het om gaat, gevolgd door een korte beschrijving van deze term (op dezelfde regel).

Deze beschrijving wordt gevolgd door een lijst met parameters die bij het uitvoeren van het commando kunnen worden gebruikt. Optionele parameters staan tussen haakjes, verplichte parameters niet.

De informatie over elke parameter begint met de naam van de parameter, gevolgd door een beschrijving van de gegevensstructuur. Ten slotte wordt een korte beschrijving van de parameter gegeven.

De definitie van een commando bestaat uit een korte beschrijving van wat het commando doet (de onderstreepte tekst) en de syntaxis. De syntaxis is de vorm waarin u het commando in een script moet gebruiken. De syntaxis kan uit verschillende delen bestaan.

Woorden die vet worden weergegeven, zijn de termen die u in een instructie gebruikt. U moet deze termen precies zo schrijven als ze in het woordenboek staan.

Woorden die normaal worden weergegeven, zijn gegevens die u opgeeft. Dit kan bijvoorbeeld een object zijn of een variabele die u hebt gedefinieerd.

Gedeelten tussen haakjes zijn optioneel. Deze hoeven niet te worden opgenomen in een instructie.

Commando's hebben betrekking op objecten. De definitie van een object bestaat uit de syntaxis die de identificatie van het

object vormt, inclusief de naam van het object en een korte beschrijving (de onderstreepte tekst). De klasse van een object is bepalend voor een aantal kenmerken van dat object, bijvoorbeeld de commando's die u op het object kunt toepassen.

VERWANTE ONDERWERPEN

Database-events
Klassendefinities
Commandodefinities
Objectdefinities
Informatie over de manier waarop AppleScript werkt

Objectdefinities

Commando's hebben betrekking op objecten. De definitie van een object in een AppleScript-woordenboek bestaat uit de syntaxis die de identificatie van het object vormt, inclusief de naam van het object en een korte beschrijving (de onderstreepte tekst). De klasse van een object is bepalend voor een aantal kenmerken van dat object, bijvoorbeeld de commando's die u op het object kunt toepassen.

Als een object een meervoudsvorm heeft, kunt u in een script naar meerdere objecten van hetzelfde type verwijzen.

Een object kan een lijst met elementen bevatten. Dit zijn andere objecten die zich in het object bevinden. De naam van een element kan in het script worden gebruikt en wordt vet weergegeven. De normale tekst die op de naam van het element volgt, beschrijft hoe in het script naar het element moet worden verwezen.

Objecten kunnen kenmerken ("property's") hebben. De naam van een kenmerk wordt met vette blauwe letters weergegeven. De normale blauwe tekst die op de naam van het kenmerk volgt, beschrijft het type waarde dat het kenmerk bevat, bijvoorbeeld tekst (een "string"). Kenmerken kunnen een constante waarde of een reeks constante waarden hebben. Een script kan de waarde van een kenmerk opvragen en instellen. Instellen is niet mogelijk als de syntaxis van het kenmerk wordt gevolgd door '[r/o]' ('read-only' ofwel alleen lezen).

VERWANTE ONDERWERPEN

AppleScript gebruiken om opdrachten te geven aan programma's
Commandodefinities
Klassendefinities
Opdrachten aan objecten geven
AppleScript-woordenboeken gebruiken

OPEN HET VOLGENDE

AppleScript Language Guide: Class Reference
AppleScript Language Guide: Kenmerken

If-instructies

Een if-instructie is een commando die alleen worden uitgevoerd als aan bepaalde voorwaarden wordt voldaan. Elke if-instructie bevat een booleaanse expressie die waar (true) of onwaar (false) is. De commando's in de if-instructie worden alleen uitgevoerd als de waarde van de booleaanse expressie waar is. De if-instructie wordt afgesloten met een end if-instructie. Bijvoorbeeld:

```
if ontvangenBedrag > 100 then
	set lidStatus to "speciaal"
end if
```

Als de booleaanse expressie in bovenstaande instructie waar is (het ontvangen bedrag is hoger dan 100), worden de commando's binnen de if-instructie uitgevoerd. Als de booleaanse expressie niet waar is, worden de commando's niet uitgevoerd.

De if-instructie kan ook worden samengesteld met een 'else'-instructie. Dit wordt een if-then-else-instructie genoemd. Bijvoorbeeld:

```
if ontvangenBedrag > 100 then
	set lidStatus to "speciaal"
else
	set lidStatus to "algemeen"
end if
```

Als de eerste booleaanse expressie in bovenstaande instructie waar is (het ontvangen bedrag is hoger dan 100), worden de commando's binnen het eerste gedeelte van de if-instructie uitgevoerd. Als de eerste booleaanse expressie niet waar is, worden de commando's in het tweede gedeelte van de instructie na 'else' uitgevoerd.

U kunt if-instructies in elkaar nesten zodat u meerdere voorwaarden kunt gebruiken.

Raadpleeg de AppleScript Language Guide (Engelstalig) voor meer informatie over if-instructies in AppleScript.

VERWANTE ONDERWERPEN

Repeat-instructies
Commandodefinities
Variabelen instellen in AppleScript
Constante waarden

OPEN HET VOLGENDE

AppleScript Language Guide - If

Repeat-instructies

U kunt repeat-instructies gebruiken om taken meerdere malen uit te voeren, afhankelijk van bepaalde voorwaarden. Met een repeat-instructie wordt een groep instructies in een lus uitgevoerd totdat aan een bepaalde voorwaarde wordt voldaan en de lus wordt verlaten. U kunt repeat-instructies in elkaar nesten, zodat op verschillende voorwaarden kan worden gereageerd.

AppleScript bevat diverse repeat-instructies, die op verschillende voorwaarden kunnen reageren.

---

repeat (forever)

De instructies worden continu herhaald in een lus. Gebruik een exit-instructie om de lus te verlaten.

```
repeat
	-- instructie
```

```
        -- instructie
        if (x = 5) then exit repeat
end repeat
```

---

repeat (aantal) times

De instructies worden het opgegeven aantal keren herhaald.

```
repeat 5 times
        -- instructie
        -- instructie
end repeat
```

---

repeat while

De instructies worden herhaald zolang aan een opgegeven voorwaarde wordt voldaan.

```
set x to 1
repeat while (x < 5)
        -- instructie
        -- instructie
        set x to (x + 1)
end repeat
```

---

repeat until

De instructies worden herhaald totdat aan een opgegeven voorwaarde wordt voldaan.

```
set x to 1
repeat until (x = 5)
        -- instructie
        -- instructie
        set x to (x + 1)
end repeat
```

---

repeat with (variabele) from (beginwaarde) to (eindwaarde)

De variabele wordt ingesteld op de beginwaarde en vervolgens na elke lus opgehoogd. De repeat-instructie wordt beëindigd wanneer de variabele de eindwaarde heeft bereikt.

```
repeat with x from (5) to (25)
        -- instructie
        -- instructie
end repeat
```

---

repeat with (variabele) in (lijst)

De variabele wordt ingesteld op het eerste item in de lijst. Na elke lus wordt de variabele ingesteld op het volgende item. De repeat-instructie wordt beëindigd wanneer alle items in de lijst zijn gebruikt.

```
set mijnLijst to {234, 342, 681, 874}
repeat with x in mijnLijst
        -- instructie
        -- instructie
end repeat
```

Raadpleeg de AppleScript Language Guide (Engelstalig) voor meer informatie over repeat-instructies in AppleScript.

VERWANTE ONDERWERPEN

If-instructies
Variabelen instellen in AppleScript
Constante waarden

OPEN HET VOLGENDE

AppleScript Language Guide - repeat

Constante waarden

Een constante is een gereserveerd woord in AppleScript dat een vooraf bepaalde waarde bevat. AppleScript bevat een groot aantal constanten die u kunt gebruiken om informatie over objecten op te vragen, vergelijkingen te maken of rekenkundige bewerkingen uit te voeren. Programma's en andere objecten kunnen hun eigen constanten definiëren. Deze constanten worden gedefinieerd in de waardenklassendefinities.

Om de waarde van een constante te zien, plakt u het volgende voorbeeld in AppleScript-editor en klikt u vervolgens op 'Voer uit'. Klik op 'Resultaat' om de waarde van de constante weer te geven.

Het resultaat is ook een constante. Dit is een scriptconstante die naar de waarde verwijst die door een commando is geretourneerd.

```
get the current date
```

Andere voorbeelden van constanten zijn:

- Rekenkundig: pi
- Booleaans: true, false
- Datum en tijd: minutes, hours, days, weeks
- Tekenreeksen: return, space, tab
- Scripts: anything, current application, it, me, missing value, my, result

Raadpleeg de AppleScript Language Guide (Engelstalig) voor meer informatie over constanten in AppleScript.

VERWANTE ONDERWERPEN

Variabelen instellen in AppleScript
Objectdefinities
Klassendefinities
Eventdefinities

OPEN HET VOLGENDE

AppleScript Language Guide - constants

Elementaire commando's

Variabelen instellen in AppleScript

Een variabele is een container die binnen een script wordt aangemaakt en wordt gebruikt voor het bewaren van een waarde. Bij het aanmaken van de variabele geeft u een naam, of ID, op. Als u een script uitvoert, kunt u een waarde in de variabele plaatsen, de waarde openen of deze wijzigen. Variabelen blijven alleen tijdens de uitvoering van het script behouden. Zodra het script wordt gesloten, gaat de waarde verloren.

- In deze voorbeelden ziet u hoe u een unieke variabele opgeeft en er een waarde aan toekent.

```
set mijnVar to "153"


set x to "Kwartaalverslagen"


set x to false
```

Om te zien hoe waarden in een variabele worden bewaard, plakt u deze voorbeelden in AppleScript-editor en klikt u vervolgens op 'Voer uit'. Klik op 'Resultaat' om de waarde van de variabele weer te geven.

Raadpleeg de AppleScript Language Guide (Engelstalig) voor meer informatie over variabelen in AppleScript.

VERWANTE ONDERWERPEN

Constante waarden
If-instructies
Repeat-instructies

OPEN HET VOLGENDE

AppleScript Language Guide - variables

Elementaire commando's

De locatie van een programma opgeven

Programma's in OS X zijn gebundeld als "pakketten". Een pakket is een speciale map die alle mappen en bestanden bevat waaruit het programma bestaat. De commando's 'path to' en 'info for' kunnen daarom voor een programma afwijkende resultaten opleveren.

Als u het commando 'path to' gebruikt voor een programma, eindigt het pad gewoonlijk op '.app'. Bestandsextensies worden mogelijk niet in de Finder weergegeven als onderdeel van een bestandsnaam, maar u kunt deze extensies wel in een script gebruiken om een bestand te identificeren.

Als u het commando 'info for' gebruikt om informatie over een programma op te vragen, heeft het kenmerk 'folder' in het resultaat gewoonlijk de waarde 'true'. Voor meer informatie over het bestand en de bestandsnaam kunt u gebruikmaken van extra kenmerken in het resultaat, zoals 'extension hidden' (verborgen extensie), 'name extension' (naamextensie) of 'displayed name' (weergegeven naam).

VERWANTE ONDERWERPEN

path to
info for
Verwijzen naar mappen, bestanden en schijven
choose application
Finder-scriptcommando's

Finder-commando's

Verwijzen naar mappen, bestanden en schijven

In AppleScript kunt u naar bepaalde mappen, bestanden en schijven verwijzen door alleen de naam ervan op te geven. U hoeft geen pad in te voeren. AppleScript kent de locatie van onderdelen als 'Documents folder', 'desktop', 'home folder' en 'startup disk'.

- In dit voorbeeld ziet u hoe u naar een speciale map verwijst door de naam ervan in een script te gebruiken.

```
set x to the path to the desktop folder
```

VERWANTE ONDERWERPEN

In scripts verwijzen naar de Finder
path to
info for
Finder-scriptcommando's

Finder-commando's

In scripts verwijzen naar de Finder

Als u commando's voor de Finder wilt gebruiken, stuurt u deze commando's met behulp van de instructie 'tell' naar de Finder. Dit is vergelijkbaar met het sturen van commando's naar een programma.

- Hier ziet u een voorbeeld van het gebruik van de instructie 'tell' in een script. De tell-instructie gebruikt de functie voor het opvragen van het aantal vensters voor Finder.app.

```
tell application "Finder.app"
        return the number of windows
end tell
```

VERWANTE ONDERWERPEN

Finder
Opdrachten aan objecten geven
Verwijzen naar mappen, bestanden en schijven
Objectdefinities
Finder-scriptcommando's

OPEN HET VOLGENDE

Finder-scriptcommando's

Er zijn verschillende commando's die u met de Finder kunt gebruiken, zoals:

activate

alias file

alias list

application file

clean up

close

column

Computer-object

container

copy

count

data size

delete

desktop window

desktop

disk

document file

duplicate

eject

empty trash

exists

file

file information

find

Finder

Finder window

folder

get

icon view options

information window

Internet location file

item

list view options

make

move

moving folder window for

open

package

path to

preferences

preferences window

print

quit

Verwijzen naar mappen, bestanden en schijven

In scripts verwijzen naar de Finder

reveal

select

set

sort

trash

update

window

Scriptextra-commando's

De scriptextra's voegen een groot aantal handige commando's toe die u in uw scripts kunt opnemen.

adding folder items to

beep

choose application

choose color

choose file

choose file name

choose folder

choose from list

choose URL

clipboard info

close access

closing folder window for

current date

delay

dialog reply

display

display dialog

do shell script

download

FTP item

get eof

info for

list disks

list folder

load script

mount volume

offset

open for access

open location

opening folder

path to

POSIX path

random number

read

round

run script

say

scripting components

set eof

set the clipboard to

set volume

store script

summarize

system attribute

time to GMT

upload

URL

write